REMARKS

Claims 1-64 are pending. Claims 1-64 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent Application Publication No. 2003/0163479 to Mathews et al. in view of U.S. Patent Application Publication No. 2003/0140027 to Huttel et al.

Reconsideration is requested. The rejections are traversed. No new matter is added. Claims 1, 10-11, 17, 21-22, 24, 26-28, 35, 37-38, 42-43, 50, 53-54, and 57-58 are amended. Claims 1-64 remain in the case for consideration.

CLAIM REJECTIONS 35 U.S.C. § 103(a)

Claim 1 is directed toward a computer, comprising: a memory; and a data structure for a generic document, the data structure stored in the memory and including: a definition of a first element, the definition of the first element including an element value field; a key identifier to identify a key value field to be used as a key in a data store.

Claim 11 is directed toward a computer system, comprising: a data store to store a first generic document; and a first schema applicable to the first generic document, the first schema including: a definition of a first element, the definition of the first element including an element value field; a first key identifier to identify a first key value field in the first generic document to be used as a key in a data store.

Claim 27 is directed toward a method for adding a generic document to a data store in a computer, comprising: accessing a schema for the generic document; locating a key value field in the generic document defined in the schema as a key; loading a value from the key value field; storing the value in the data store; and storing the generic document in the data store.

Claim 43 is directed toward a method for defining a schema for a generic document in a computer, comprising: defining a first element in the schema, the first element including an element value field; and identifying a key value field in the schema to be used as a key in a data store.

Claim 50 is directed toward a computer-readable medium containing a program to add a generic document to a data store, comprising: software to access a schema for the generic document; software to locate a key value field in the generic document defined in the schema as a key; software to load a value from the key value field; software to store the value in the data store; software to store the generic document in the data store; and software to index the generic document in the data store using the value.

Claim 58 is directed toward a computer-readable medium containing a program to define a schema for a generic document, comprising: software to define a first element in the schema, the first

element including an element value field; and software to identify a key value field in the schema to be used as a key in a data store.

In contrast, Matthews teaches a method and apparatus for implementing a data management system. An object relationship model is created automatically using a specification of the data storage schema. The object relationship model defines views that limit how consumers can interact with the data store and validate that the data store is being used properly. Consumers interact with the data store through the object relationship model through the views.

The Examiner has only cited to Matthews in rejecting claims 11, 27, and 43. The Examiner did not cite to Huttel in rejecting claims 11, 27, and 43. As claim 11, 27, and 43 are rejected under 35 U.S.C. § 103(a) as unpatentable over Matthews in view of Huttel, the Applicant asserts that Huttel does not teach any of the features of claims 11, 27, and 43. The Examiner has rejected claim 50-64 on the same rationale as claim 1-49, without any specificity.

The Applicant notes that the Examiner has found a reference that uses certain key terms found in the claims, such as "document", "data store", and "schema". But the Applicant respectfully suggests that the Examiner has misunderstood the invention, and that Matthews does not make the invention obvious.

The claimed invention is concerned with adding a generic document to a data store, but in a way that provides some information about the contents of the generic document. To accomplish this, a schema is accessed for the document. The schema is used to locate a key value field in the generic document. The located value can then be loaded and stored in the data store, along with the generic document itself.

There are two points worth noting about the claimed invention. First, the schema or data structure relates to the generic document, not the data store. Second, more data is stored in the data store than just the document. These points are discussed in turn below.

First, claim 1 recites "a data structure for a generic document", claim 11 recites "a first schema applicable to the first generic document," claims 27 and 50 recite "accessing a schema for the generic document", and claims 43 and 58 recite "defining a schema for a generic document." The schema of Matthews is a "data storage schema" (*see* Matthews, ¶ 9). All other references to "schema" in Matthews are explicitly or implicitly schemas for the "data store" or "database". The data store schema of Matthews is used only by the Data Access Controller (DAC) to generate the Application Programming Interface (API) and to validate data use. In other words, the schema of Matthews is not used to do anything in relation to a document; it is used to define how to access data from a data store. Matthews merely uses the schema to define the APIs that permit a consumer to

access data from the data store. Because Matthews teaches a schema only for the "data store" or the "database", Matthews does not teach or suggest a schema "for the generic document". Further, a generic document and a database are different objects, and the schemas of each are distinct. It is worth noting that the independent claims all recite both the "generic document" and the "data store", and that independent claims 11, 27, 43, 50, and 58 all describe the "schema" as applying to or being used for a "generic document". Thus, the data store schema of Matthews does not teach any limitations regarding a schema for a generic document. As a result, Matthews cannot make obvious independent claims 1, 11, 27, 43, 50, and 58.

The Applicant notes that "a schema is a definition of the format a document should take if it is to satisfy the schema" (*see* specification, page 4, lines 20-21). This definition is fairly consistent with a schema that defines a data store. But a schema for a generic document enables different document types to have a different schema. In contrast, a single database, such as those of Matthews, will have only a single schema. The schema defines the structure of the data store, and applies only to that one data store. The schema is not usable with another data store (unless the data store happens to have an identical structure). To access data from a generic document, the appropriate schema is used. As the schema of Matthews defines the data store and not the document, it cannot be used to access data from the document.

Second, the purpose of the schema is to support adding the generic document, along with other data, to the data store. For example, claim 11 recites "a data store to store a first generic document", and "a first schema applicable to the first generic document...including a first key identifier to identify a first key value field in the first generic document to be used as a key in a data store." Similarly, claims 27 and 50 recite "locating a key value field in the generic document defined in the schema as a key", "loading a value from the key value field; storing the value in the data store", and "storing the generic document in the data store". In other words, the schema is used to retrieve a value from the generic document; that value and the generic document are both stored in the data store. Nothing in Matthews teaches similar functionality. As said above, the schema in Matthews is not used to access data from a document, it is used to define how to access data from a data store.

In rejecting claims 27 and 50, the Examiner points to ¶ 73 as teaching that objects can be added to the data store. While Matthews might teach adding objects to the data store, the Examiner has missed the point of the claim. Claims 27 and 50 recite storing two items in the data store: the generic document itself, and a value loaded from a key value field in the generic document, located using the schema. While Matthews might describe adding objects generally, that does not mean that

Matthews teaches adding a generic document and a value retrieved from that document, as claimed. Matthews does not teach or suggest the possibility of adding both an object and data derived from the object. Even more, Matthews does not teach or suggest how the data could be derived from the object in the case of claims 27 and 50, the value loaded from the field in the document.

To further illustrate the differences in the art with respect to the present invention, the Applicant notes that the claimed invention complements Matthews and Huttel. For example, one could implement the schema in Matthews to define the APIs that permits a consumer to access data from the data store and achieve interoperability with other systems. In conjunction, one could implement the claimed invention to enable the data store to organize generic documents using the schemas for the documents.

Because neither Matthews nor Huttel teaches or suggests using a schema for a generic document, claims 1, 11, 27, 43, 50, and 58 are patentable are patentable under 35 U.S.C. § 103(a) over Matthews in view of Huttel. Additionally, because neither Matthews nor Huttel teach or suggest loading a value from the generic document using the schema, or storing the generic document and the value in the data store, claims 11, 27 and 50 are patentable over Matthews in view of Huttel. Accordingly, claims 1, 11, 27, 43, 50, and 58 are patentable, as are dependent claims 2-10, 12-26, 28-37, 44-49, 51-54, and 59-64.

Claim 17 is directed toward a system according to claim 11, wherein: the data store is operative to store a second generic document; and the system further comprises a second schema applicable to the second generic document, the second schema including: a definition of a second element, the definition of the second element including an element value field; a second key identifier to identify a second key value field in the second generic document to be used as a key in a data store.

Claim 18 is directed toward a system according to claim 17, wherein: the first schema includes a first identifier for the first key value field; the second schema includes a second identifier for the second key value field; and the first identifier and the second identifier are the same identifier.

In rejecting claims 17 and 18, the Applicant notes that the Examiner did not even address the fact that claims 17 and 18 both recite "a second generic document" and "a second schema applicable to the second generic document." As argued above, Matthews teaches only a single schema for a single database. Matthews does not teach a schema for even a single generic document, let alone two generic documents. The Examiner did not cite to Huttel as teaching this feature, and the Applicant asserts that Huttel does not teach or suggest this feature either. As neither Matthews nor Huttel

teaches or suggests two schemas for two generic documents, claims 17 and 18 are patentable under 35 U.S.C. § 103(a) over Mathews in view of Huttel. Accordingly, claims 17 and 18 are allowable.

Claim 38 is directed toward a method for defining a data store in a computer, comprising: accessing a schema; locating an object defined in the schema as a key; defining a first data structure in the data store for the object; identifying the first data structure in the data store as a key data structure; and defining a second data structure in the data store for a generic document conforming to the schema.

Claim 55 is directed toward a computer-readable medium containing a program to define a data store, comprising: software to access a schema; software to locate an object defined in the schema as a key; software to define a first data structure in the data store for the object; software to identify the first data structure in the data store as a key data structure; and software to define a second data structure in the data store for a generic document conforming to the schema.

First, the arguments presented above with reference to claims 1, 11, 27, 43, 50, and 58 apply equally to claims 38 and 55, for the same reasons. The "schema" of claims 38 and 55 is distinguishable from the schema of Matthews, because it relates to a generic document, and not to the data store.

Second, in rejecting claims 38-42, the Examiner makes a blanket rejection, identifying only particular features of the claims in question. The Examiner further rejects claims 50-64 under the same rationale as claims 1-49 without further specificity.

The Applicant believes that there are at least two features of independent claims 38 and 55 that distinguish these claims over the prior art. Specifically, claim 38 recites "defining a first data structure in the data store for the object", and "defining a second data structure in the data store for a generic document conforming to the schema." Similarly claim 55 recites "software to define a first data structure in the data store for the object", and "software to define a second data structure in the data store for a generic document conforming to the schema."

Because the Examiner does not specifically discuss these features of claims 38 and 55, the Applicant reviewed the entirety of Matthews and Huttel, to see if anything in the references was pertinent. The Applicant did not find any teachings in either Matthews or Huttel that seemed pertinent to the rejection of claims 38 and 55. As argued above, the schema of the claimed invention relates to the generic document, not the data store, and more data is stored in the data store than just the document.

The closest teaching the Applicant could find in either Matthews or Huttel that seemed pertinent to claims 38 and 55 was ¶ 0053 of Matthews, where Matthews states that "the generated data store schema defines a data structure using the OASM specification". But as argued above, Matthews merely uses the schema to define the APIs that permit a consumer to access data from the data store. Thus, Matthews does not teach or suggest defining structures in the data store for an object defined in a schema as a key or defining a data structure in the data store for a generic document conforming to the schema.

As neither Matthews nor Huttel teach or suggest defining a data structure in a data store, claims 38 and 55 are patentable under 35 U.S.C. § 103(a) over Matthews in view of Huttel. Accordingly, claims 38 and 55 are allowable, as are dependent claims 39-42 and 56-57.

For at least the foregoing reasons, reconsideration and allowance of claims 1-64 of the application as amended is requested. The Examiner is encouraged to telephone the undersigned at (503) 222-3613 if it appears that an interview would be helpful in advancing the case.

Respectfully submitted,

MARGER JOHNSON & McCOLLOM, P.C.

Ariel S. Rogson
Reg. No. 43,054

MARGER JOHNSON & McCOLLOM, P.C.
210 SW Morrison Street, Suite 400
Portland, OR 97204
503-222-3613
Customer No. 45842